IMAGE CAPTION GENERATOR USING CNN AND LSTM(GUI Application)

Rohit Pawar, Omini Jadhav, Rutuja Nalage

SVPM's College of Engineering Malegaon (BK), Baramati, Maharashtra, India

(Prof Khalate. Y.R, Department of Computer Engineering, SVPM's College of Engineering Malegaon (BK), Baramati, Maharashtra, India)

ABSTRACT

Image captioning is one of the most needed requirements of today's world. Moreover, there are some inbuilt applications that generate and provide a caption for a certain image, all these things are done with the help of deep neural network models. The process of generating a description of an image or the process by which we get to know moto of image is called image captioning. Using CNN it detects the important objects, their attributes and also relation among them in respective image. It generates syntactically and semantically correct sentences. In this project, we present a CNN LSTM Model to describe images and generate captions using computer vision and machine translation. This project aims to detect different and important objects found in an image, recognize the relationships between those objects and predicting the sequence of sentence using LSTM. The dataset used is Flickr8k and the programming language used is Python3, and with the help of the Xception model, to demonstrate the caption for image. This project will also elaborate on the functions and structure of the various Neural networks involved. Generating image captions is an important aspect of Computer Vision and Natural language processing.

Keywords: CNN, LSTM, BLEU, VGG16, Image captioning, deep learning.

I. INTRODUCTION

In this Project we will be predicting the contents of an image by words one by one and make a sensible sentence to be able to describe this image. A network of CNN and LSTM is used in this project. CNN is used to extract features and LSTM to store the words one by one and make a sentence. The caption should not only be able to describe the object but also make a sensible sentence which describe the action that's going on in that image .Most of the previous methods are based on indexing and labelling an image and categorizing them which can be a huge waste of human efforts when being compared to the automatic generation through deep learning .However simple words in any language can be feeded to the model to be able to predict the words of the objects In an image[5] Every day we see a lot of photographs in the surroundings , on social media and in the newspapers. Humans are able to recognize photographs themselves only. We humans can pick out the photographs without their designated captions but on the other hand machines need images to get trained first then it'd generate the photograph caption automatically. Image captioning may benefit for loads of purposes, for example supporting the visionless person using text-to-speech through real time feedback about encompassing the situation over a camera feed, improving social medical leisure with the aid of reorganizing the captions for photographs in social feed along with messages to speech.

II. METHODOLOGY

We are using CNN to detect different objects in given image and LSTM to predict the sequence of words and in turn generate the caption for given Image.

CNN

CNN is used for extracting features from the image. CNN- Convolutional Neural networks are specialized deep neural networks which can process the data that has input shape like a 2D matrix. Images are easily represented as a 2D matrix and CNN is very useful in working with images. CNN is basically used for image classifications and identifying if an image is a bird, a plane or Superman, etc. It scans images from left to right and top to bottom to pull out important features from the image and combines the feature to classify images. It can handle the images that have been translated, rotated, scaled and changes in perspective.

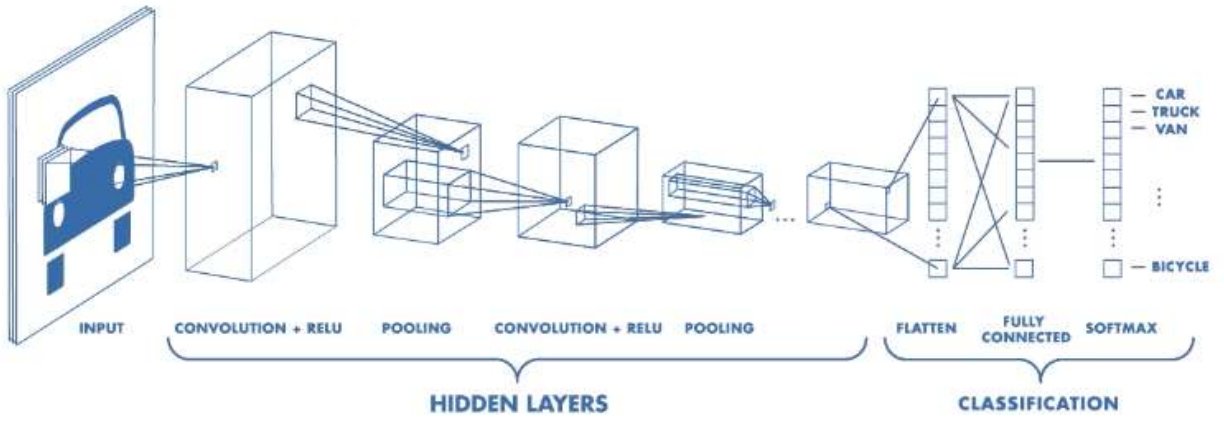


Fig. CNN Architecture

LSTM

LSTM stands for Long short-term memory; they are a type of RNN (recurrent neural network) which is well suited for sequence prediction problems. Based on the previous text, we can predict what the next word will be. It has proven itself effective from the traditional RNN by overcoming the limitations of RNN which had short term memory. LSTM can carry out relevant information throughout the processing of inputs and with a forget gate, it discards non-relevant information. LSTM will use the information from CNN to help generate a description of the image

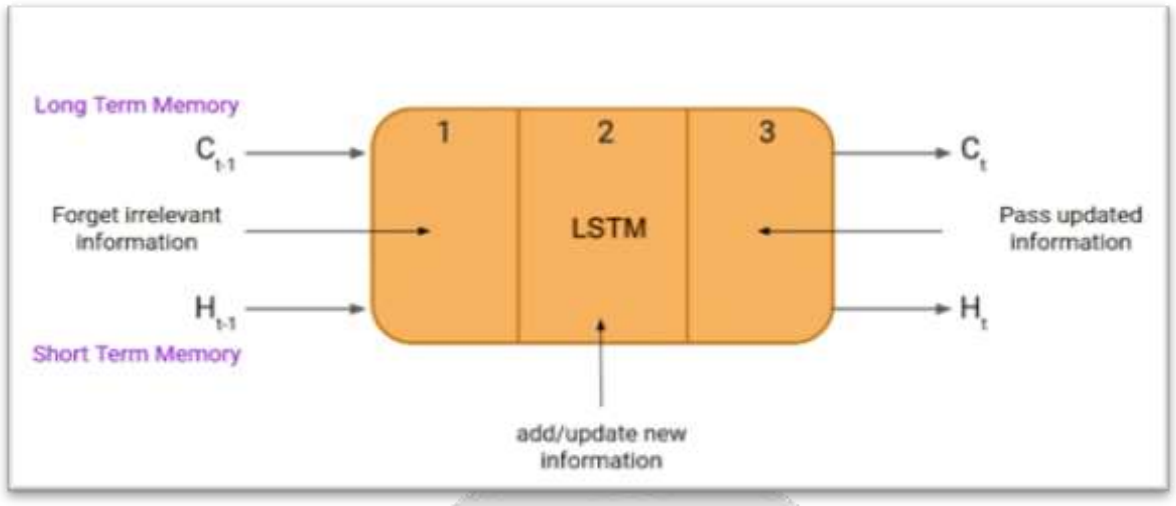


Fig. LSTM Gates

Forget Gate: -

In a cell of the LSTM network, the first step is to decide whether we should keep the information from the previous timestamp or forget it. So forgot get is used for that purpose.

Forget Gate:

- $f_t = \sigma (x_t * U_f + H_{t-1} * W_f)$

Later, a sigmoid function is applied over it. That will make ft a number between 0 and 1. This ft is later multiplied with the cell state of the previous timestamp as shown below.

$C_{t-1} * f_t = 0$ ...if $f_t = 0$ (forget everything)

$C_{t-1} * f_t = C_{t-1}$ ...if $f_t = 1$ (forget nothing)

If ft is 0 then the network will forget everything and if the value of ft is 1 it will forget nothing.

Input Gate: -

Input gate is used to quantify the importance of the new information carried by the input. Here is the equation of the input gate.

Input Gate:

- $i_t = \sigma (x_t * U_i + H_{t-1} * W_i)$

Again, we have applied sigmoid function over it. As a result, the value of I at timestamp t will be between 0 and 1.

New Information: -

- $N_t = \tanh(x_t * U_c + H_{t-1} * W_c)$ (new information)

Now the new information that needed to be passed to the cell state is a function of a hidden state at the previous timestamp t-1 and input x at timestamp t. The activation function here is tanh. Due to the tanh function, the value of new information will between -1 and 1. If the value is of Nt is negative the information is subtracted from the cell state and if the value is positive the information is added to the cell state at the current timestamp.

However, the Nt won't be added directly to the cell state. Here comes the updated equation

$C_t = f_t * C_{t-1} + i_t * N_t$ (updating cell state)

Output Gate: -

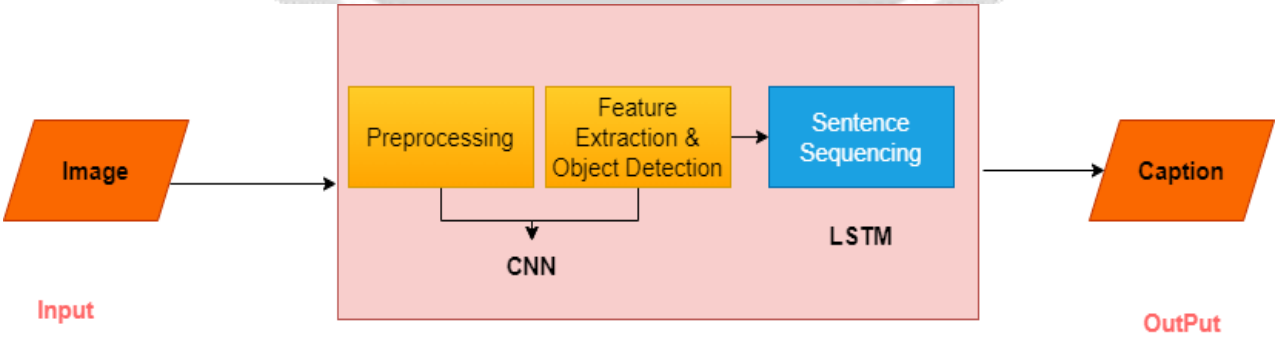
The output gate determines the value of the next hidden state. This state contains information on previous inputs. First, the values of the current state and previous hidden state are passed into the third sigmoid function. Then the new cell state generated from the cell state is passed through the tanh function. Both these outputs are multiplied point-by-point. Based upon the final value, the network decides which information the hidden state should carry. This hidden state is used for prediction.

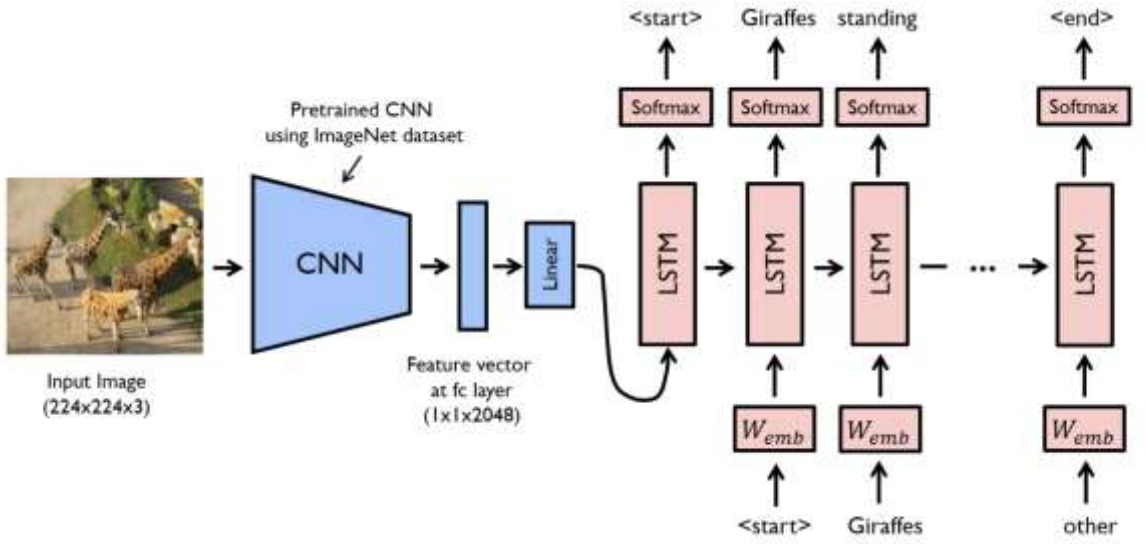
Output Gate:

- $o_t = \sigma(x_t * U_o + H_{t-1} * W_o)$

$H_t = o_t * \tanh(C_t)$

II. SYSTEM ARCHITECTURE





Three Phases of the application

- Features Extraction: The extraction of the features from images are being extracted. It creates vector features also known as embeddings. The CNN model extract features from original images after which they are compressed to smaller and RNN compatible feature vector.
- Tokenization: The next phase in the application is RNN, that decodes the feature vectors that were fed to it from CNN. Here the sequence of the words is predicted and however the captions are generated.
- Prediction: After the tokenization, the last step is Prediction.

LITERATURE SURVEY: -

Technology Used	IEEE Paper Name	Algorithm/ Library Used	Advantages	Disadvantages
Deep Learning [3]	Camera2Caption: A Real-Time Image Caption Generator [Pranay Mathur∗, Aman Gill†, Aayush Yadav‡, Anurag Mishra§ and Nand Kumar Bansode]	Convolutional Neural Network (CNN). Long Short Memory(LSTM)	1. Generating well -formed sentence requires both syntactic and semantic of the languaguage. 2. Computationally efficient. 3. Very High accuracy in image recognition problems.	1. CNN do not encode the position and orientation of object. 2. Lack of ability to be spatially invariant to the input data. 3.Lots of training data is required.
Deep Learning[5] [4]	Image Caption Generator Using CNN [K. Praveen Kumar1 , V. Prakash Reddy2 G. Indra Karan Reddy3, N.S. Ganesh4] Image Caption Generator Using CNN and LSTM[Swarnim Tripathi,Ravi Sharma]	Convolutional Neural Network (CNN). Long Short Memory(LSTM)	1. Generate Caption Using CNN and LSTM. 2. It uses resnet 50 model to extract feature. 3.uses CNN for feature extraction and LSTM to predicting the next words.	1. Accuracy issue is there.

Deep Neural Network [2]	Image Caption Generator Based On Deep Neural Networks [Jianhui Chen Wenqiang Dong, Minchen Li]	Convolutional Neural Network (CNN). Long Short Memory(LSTM). Recurrent neural network(RNN)	1. Easy to use. 2. Provides inbuilt things.	1. Problems in low-level API.
Deep Learning[6]	Show and Tell: A Neural Image Caption Generator [Oriol Vinyals Google ,Alexander Toshev Google Samy Bengio Google ,Dumitru Erhan Google]	Convolutional Neural Network (CNN). Long Short Memory(LSTM). Recurrent neural network(RNN)	1. Reliability. 2. Accuracy. 3. Process in a simpler and faster way.	1. Failing in image processing.
Deep Learning [7]	Domain-Specific Image Caption Generator with Semantic Ontology[Seung-Ho Han and Ho-Jin Choi]	Convolutional Neural Network (CNN). Long Short Memory(LSTM). Recurrent neural network(RNN)	1.Generate domain-specific image caption based on object and attribute information. 2.uses semantic ontology to provide natural language description. 3.uses moscoco data set which include general images.	1. model is not end-to-end manner for semantic ontology

III. RESULTS AND DISCUSSION

DEEP LEARNING MODEL	ACTIVATION FUNCTION	COST FUNCTION	EPOCHS	GRADIENT ESTIMATION	NETWORK ARCHITECTURE	NETWORK INITIALIZATION	Output Mean BLEU score
Gradient Estimation							
1	ReLU	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.37
2	ReLU	Cross-Entropy	6	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.351
3	ReLU	Cross-Entropy	5	Adagrad	3 layer, 256 nodes, LSTM, vgg16	default	0.404
4	ReLU	Cross-Entropy	5	RMSProp	3 layer, 256 nodes, LSTM, vgg16	default	0.374
5	ReLU	Cross-Entropy	5	Adadelta	3 layer, 256 nodes, LSTM, vgg16	default	0.355
6	ReLU	Cross-Entropy	5	Nadam	3 layer, 256 nodes, LSTM, vgg16	default	0.353
7	ReLU	Cross-Entropy	5	SGD	3 layer, 256 nodes, LSTM, vgg16	default	0.028
Cost Function							
1	ReLU	mean_squared_error	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.215
2	ReLU	hinge	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0
3	ReLU	kullback_leibler_divergence	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.373
4	ReLU	cosine_proximity	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0
Network Initialization							
1	ReLU	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	glorot_uniform	0.381
2	ReLU	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	random_uniform	0.388
3	ReLU	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	lecun_uniform	0.367
4	ReLU	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	he_uniform	0.389
5	ReLU	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	glorot_normal	0.398
Activation Function							
1	ReLU	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.374
2	tanh	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.384
3	elu	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.392
4	selu	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.363
5	linear	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.192
6	sigmoid	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.375
7	softsign	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.396
8	softplus	Cross-Entropy	5	Adam	3 layer, 256 nodes, LSTM, vgg16	default	0.381
Epochs							
1	ReLU	Cross-Entropy	3	Adam	3 layers, 256 nodes each	default	0.429
2	ReLU	Cross-Entropy	4	Adam	3 layers, 256 nodes each	default	0.394
3	ReLU	Cross-Entropy	5	Adam	3 layers, 256 nodes each	default	0.408
4	ReLU	Cross-Entropy	6	Adam	3 layers, 256 nodes each	default	0.38
5	ReLU	Cross-Entropy	7	Adam	3 layers, 256 nodes each	default	0.405
Network Architecture							
1	ReLU	Cross-Entropy	5	Adam	3 layers, 256 nodes each	default	0.407
2	ReLU	Cross-Entropy	5	Adam	3 layers, 128 nodes each	default	0.405
3	ReLU	Cross-Entropy	5	Adam	3 layers, 512 nodes each	default	0.394
4	ReLU	Cross-Entropy	5	Adam	4 layers, 256 nodes each	default	0.406
5	ReLU	Cross-Entropy	5	Adam	4 layers, 128 nodes each	default	0.386

Fig-3: Model results

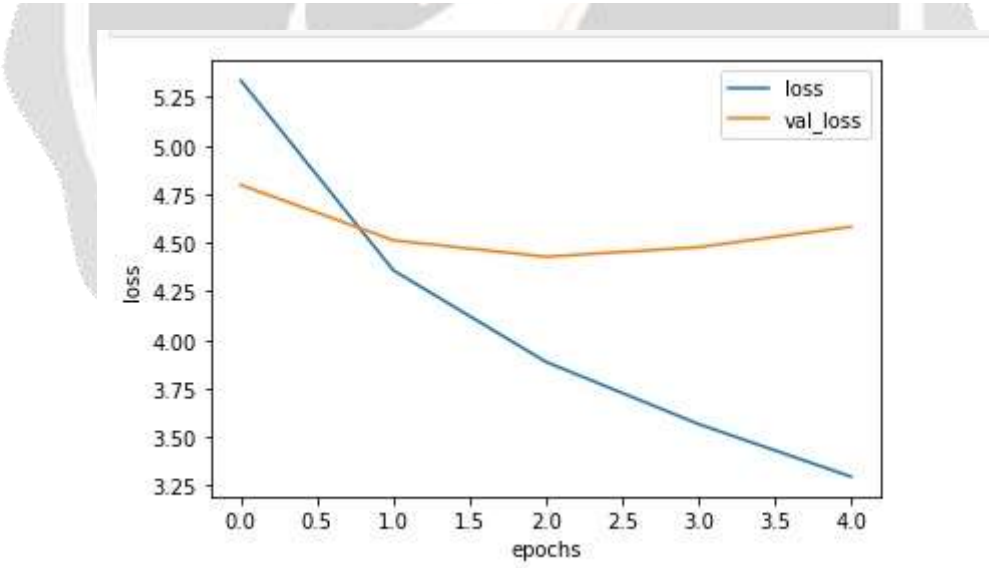


Fig-Loss Function Graph

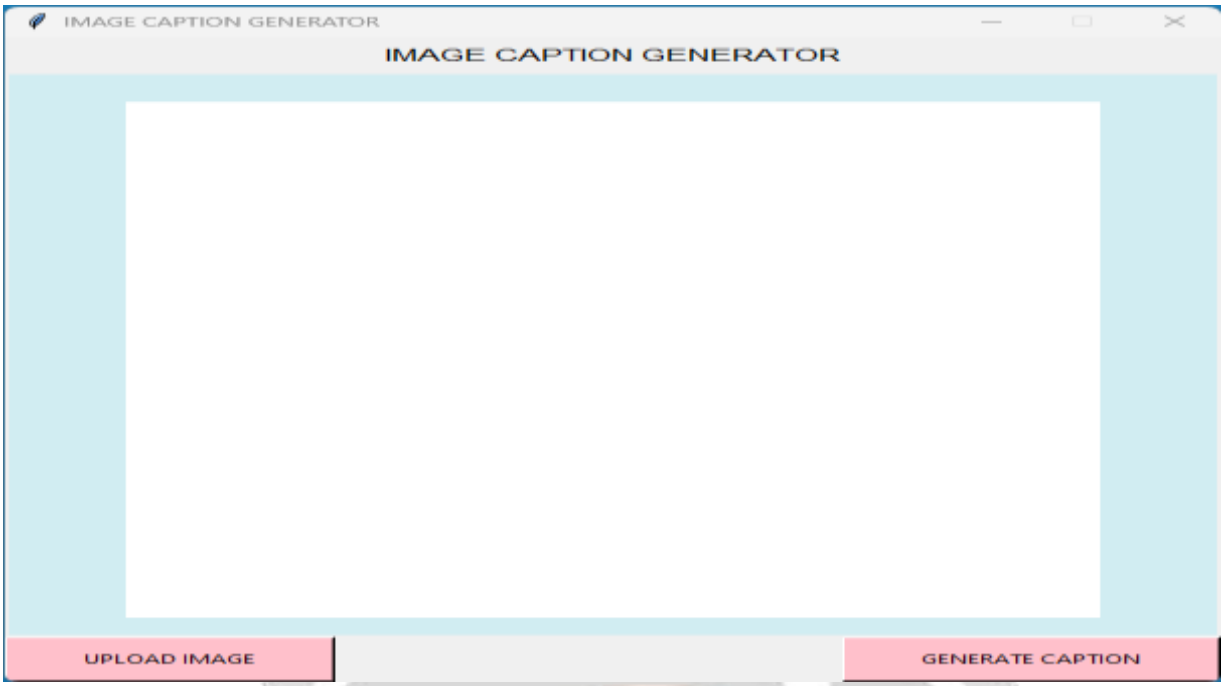


Fig-Output1

Fig-Output2

APPLICATIONS

1. In Virtual Assistants
2. Recommendations in editing
3. Image Encoding
4. Self-driving cars
5. social media
6. In Natural Language Processing Application

IV. CONCLUSION

Our model based on multi label classification using fast Text and CNN, is useful in detecting and extracting objects from image and generate caption according to the provided datasets. We have presented multiple approaches for Image caption Generator like (Convolution neural network, Long short-term memory). The CNN-LSTM model was built on the idea of generating the captions for the input pictures. This model can be used for a variety of applications. In this, we studied about the CNN model, RNN models, LSTM models, and in the end, we validated that the model is generating captions for the input pictures.

V. REFERENCES

[1] Min Yang, Junhao Liu, Ying Shen, Zhou Zhao , Member, IEEE, Xiaojun Chen , Member, IEEE, Qingyao Wu , Member, IEEE, and Chengming Li , Member, IEEE : An Ensemble of Generation- and Retrieval-Based Image Captioning With Dual Generator Generative Adversarial Network.https: //ieeexplore.ieee.org/document/9226120

[2] Jianhui Chen,Wenqiang Dong, Minchen Li : Image Caption Generator Based On Deep Neural Networks.https://www.seas.upenn.edu/~minchenl/ doc/ImgCapGen.pdf

[3] Pranay Mathur, Aman Gill†, Aayush Yadav‡, Anurag Mishra§ and Nand Kumar Bansode : Camera2Caption: A Real-Time Image Caption Generator.https://ieeexplore.ieee.org/document/8272660 [4] Swarnim Tripathi,Ravi Sharma:Image Caption Generator Using CNN and LSTM.http://ijcrt.org/papers/IJCRT_196552.pdf

[5] K. Praveen Kumar1, V. Prakash Reddy2 G. Indra Karan Reddy3, N.S. Ganesh4 : Image Caption Generator Using CNN.https://ijcrt.org/papers/ IJCRT2106298.pdf

[6] Oriol Vinyals Google,Alexander Toshev Google,Samy Bengio Google,Dumitru Erhan Google : Show and Tell: A Neural Image Caption Generator.https: //arxiv.org/abs/1411.4555

[7] Seung-Ho Han and Ho-Jin Choi :Domain-Specific Image Caption Generator with Semantic Ontology.https://ieeexplore.ieee.org/abstract/document/ 9070680/

[8]Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu :BLEU: a Method for Automatic Evaluation of Machine Translation.https://aclanthology. org/P02-1040.pdf [9] Dataset: https://www.kaggle.com/datasets/khanrahim/flickr8k